IN THE CLAIMS

1. [Cancelled]

2. [amended herein] A method according to claim ~~1~~8~~, characterized in that in that it further comprises the steps of~~ further comprising:

- when processing at least one captured data packet, determining ~~(304, 610) a~~ modification command affecting ~~at least~~ said at least one captured data packet, and
- maintaining ~~(305, 610)~~ a list of modification commands, said list enabling modification of captured data packets.

3. [amended herein] A method according to claim 2~~, characterized in that it further comprises the steps of~~ further comprising:

- modifying ~~(306, 612)~~ captured data packets based on said list of modification commands, and
- releasing ~~(308, 614)~~ modified captured data packets.

4. [amended herein] A method according to claim ~~1~~8, further comprising the step of discarding ~~characterized in that~~ captured data packets that are declined from processing ~~are discarded (408, 604)~~.

5. [amended herein] A method according to claim ~~1~~8, ~~characterized in that~~ wherein captured data packets that are declined from processing are delayed ~~(410, 603)~~.

6. [amended herein] A method ~~according to claim 2, characterized in that it further~~

~~comprises the steps of~~ for handling data packets, said data packets belonging to a set of data packets, said method comprising:

- capturing data packets,
- accepting a captured data packet for processing based on said captured data packet and data packets captured prior to said captured data packet,
- when processing at least one captured data packet accepted for processing, determining a modification command affecting at least said at least one captured data packet,
- maintaining a list of modification commands, said list enabling modification of captured data packets,
- declining ~~(404) a~~ captured data packet from processing, if said captured data packet is already processed and modification commands induced by said captured data packet are already determined and maintained in said list of modification commands,
- modifying ~~(306, 612)~~ said captured data packet based on said list of modification commands, and
- releasing ~~(308, 614)~~ the modified captured data packet.

7. [amended herein] A method according to claim ~~1~~8, ~~**characterized** in that in that in that it further comprises the steps of~~ further comprising:

- declining ~~(404) a~~ captured data packet from processing, if said captured data packet is already processed, and
- releasing ~~(308)~~ the captured data packet.

8. [amended herein] A method ~~according to claim 1, **characterized** in that~~ for handling

data packets, said data packets belonging to a ~~said~~ set of data packets ~~has~~ having at least partly hierarchical structure, said method ~~further~~ comprising ~~the step of~~:

- capturing data packets,

- accepting a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet,

- said accepting comprising accepting ~~(301, 411, 414, 422, 430)~~ data packets for processing in the order specified by said at least partly hierarchical structure.

9. [amended herein] A method according to claim 8~~, characterized in~~ comprising:

- accepting ~~(411)~~ a captured data packet for processing, if data packet(s) immediately preceding said captured data packet in said at least partly hierarchical structure is(are) already processed.

10. [amended herein] A method according to claim 8~~, characterized in~~ comprising

- declining ~~(404)~~ a captured data packet from processing, if a data packet immediately preceding said captured data packet in said at least partly hierarchical structure is not yet captured.

11. [amended herein] A method according to claim 10~~, characterized in~~ comprising

- delaying ~~(410, 603)~~ data packets declined from processing,

- accepting ~~(414)~~ a delayed data packet for processing and processing ~~(302)~~ said delayed data packet, if data packet(s) immediately preceding said delayed data packet in said at least partly hierarchical structure is(are) already processed, and

- releasing ~~(308)~~ the delayed data packet.

12. [amended herein] A method according to claim 10~~, characterized in~~ comprising

- delaying ~~(410, 603)~~ data packets declined from processing,
- accepting ~~(301, 411, 414, 422, 430)~~ a captured data packet for processing,
- accepting ~~(414)~~ delayed data packet(s) for processing, if data packet(s) immediately preceding said delayed data packet(s) in said at least partly hierarchical structure is(are) accepted for processing,
- processing ~~(418, 424)~~ said delayed data packet(s) together with said captured data packet, and
- releasing ~~(308)~~ the delayed and the captured data packets.

13. [amended herein] A method according to claim 8~~, characterized in that~~ wherein said at least partly hierarchical structure is a sequence of data packets.

14. [amended herein] A method according to claim 8~~, characterized~~ wherein in that said at least partly hierarchical structure is a hierarchically structured tree.

15. [cancelled]

16. [amended herein] A method ~~according to claim 15, characterized in~~ for handling data packets, said data packets belonging to a set of data packets and forming a plurality of groups of data packets, said method comprising:

- capturing data packets,

- declining ~~(404) a~~ captured data packet belonging to a first group of data packets from processing, if all other data packets belonging to said first group of data packets are not yet captured, and delaying ~~(410)~~ said captured data packet, and

- processing ~~(424)~~ a captured data packet belonging to a first group of data packets together with delayed data packets belonging to said first group of data packets, if said captured data packet belonging to said first group and said delayed data packets belonging to said first group form a full first group of data packets.

17. [amended herein] A method according to claim ~~15~~16, ~~**characterized** in that~~ wherein said plurality of groups of data packets further belong to a set of groups having at least partly hierarchical structure, and ~~in that said method further comprises the step of:~~ further comprising:

- processing ~~(302)~~ said groups of captured data packets in the order specified by said at least partly hierarchical structure.

18. [amended herein] A method according to claim ~~2~~6, ~~**characterized** in that~~ wherein said data packets belonging to a set of data packets are first handled ~~(800)~~ in a first node of a cluster of network elements and said list of modification commands is maintained in said first node, and in that said method further comprises the step of:

- transmitting ~~(804)~~ said list of modification commands from said first node to a second node of said cluster of network elements.

19. [amended herein] A method according to claim 18, ~~**characterized** in that it further comprises the step of~~ further comprising:

- after said transmission of said list, handling ~~(806)~~ said set of data packets in said second node.

20. [amended herein] A method according to claim 19,~~ characterized in that it further comprises the steps of~~ further comprising:
- when beginning to handle said data packets, storing in said first node in a connection data structure an entry representing said set of data packets, and
- before handling said set of data packets in said second node, transmitting said entry from said first node to said second node.

21. [amended herein] A method according to claim ~~16~~,~~ characterized in that it further comprises the step of~~ further comprising:
- defining ~~(700) a~~ plurality of first pieces of information, which are to be replaced in the captured data packets with a plurality of corresponding second pieces of information, and ~~in that in~~ wherein, in the processing of captured data packets,
- said first pieces of information are searched for ~~(702)~~, and
- if a first piece of information is found, at least one modification command specifying at least the replacement of said first piece of information with a corresponding second piece of information is determined ~~(704)~~.

22. [amended herein] A method according to claim 21,~~ characterized in that~~ wherein, if the length of said first piece of information is different from the length of said corresponding second piece of information and if said first piece of information is found in payload of data packet(s), said modification command comprises instructions for

changing value of at least one header field in a data packet.

23. [amended herein] A method according to claim 26, ~~characterized in that~~ wherein a modification command comprises

- a first identifier indicating the beginning of a first piece of information in the original captured data packets, the first piece of information being subject to be replaced by a second piece of information,
- the length of the first piece of information, and
- the second piece of information.

24. [amended herein] A method ~~according to claim 23, characterized in that~~ for handling data packets, said data packets belonging to a set of data packets, said method comprising:

- capturing data packets,
- processing captured data packets;
- accepting a captured data packet for processing based on said captured data packet and data packets captured prior to said captured data packet,
- when processing at least one captured data packet accepted for processing, determining a modification command affecting at least said at least one captured data packet,
- maintaining a list of modification commands, said list enabling modification of captured data packets,

wherein a modification command ~~further~~ comprises

- a first identifier indicating the beginning of a first piece of information in the original

captured data packets, the first piece of information being subject to be replaced by a second piece of information,
- the length of the first piece of information,
- the second piece of information,
- a second identifier indicating the beginning of the second piece of information in the modified captured data packets,
- an offset between a third identifier indicating the end of the first piece of information in the original captured data packets and a fourth identifier indicating the end of the second piece of information in the modified captured data packets, and
- the length of the second piece of information.

25. [amended herein] A method according to claim ~~16~~, ~~**characterized** in that~~ wherein said data packets contain information fragments belonging to a sequence of information fragments, said method further comprising the steps of:
- processing ~~(302, 418, 424, 432, 608)~~ the information fragments of the captured data packets in the order specified by said sequence.

26. [amended herein] A method according to claim 25~~, **characterized** in that~~ wherein each information fragment of said sequence is processed only once.

27. [amended herein] A method ~~according to claim 25, **characterized** in that said method further comprises the steps of:~~ for handling data packets, said data packets belonging to a set of data packets, said method comprising:
- capturing data packets, said data packets containing information fragments belonging to

a sequence of information fragments,

- declining ~~(404)~~ a captured data packet from processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is not yet captured, and
- accepting ~~(411)~~ a captured data packet for processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is already processed, whereby each information fragment of said sequence is processed only once.

28. [amended herein] A method according to claim 25, ~~**characterized** in that~~ wherein said sequence of information fragments is a sequence of octets of data according to Transfer Control Protocol.

29. [amended] A software entity ~~(900)~~ for handling data packets, said data packets belonging to a set of data packets, said software entity comprising

- program code means ~~(901)~~ for capturing data packets,

~~**characterized** in that it further comprises:~~

- program code means ~~(904)~~ for accepting a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet.

30. [amended herein] A software entity ~~(900)~~ according to claim 29, ~~**characterized** in that it further comprises~~ further comprising:

- program code means ~~(908)~~ for maintaining a list of modification commands, said list enabling modification of captured data packets.

31. [amended herein] A software entity ~~(900)~~ according to claim 30~~, characterized in that it further comprises~~ further comprising:
- program code means ~~(910)~~ for modifying captured data packets based on said list of modification commands, and
- program code means ~~(912)~~ for releasing modified captured data packets.

32. [amended herein] A software entity ~~(920)~~ for processing data, said software entity being adapted to receive data, and said software entity comprising
- program code means ~~(902)~~ for processing received data,
- program code means ~~(906)~~ for determining a modification command affecting at least received data, as a response to processing said data, and
- said software entity being adapted to output said modification command.

33. [original] A computer program comprising program code for performing all the steps of Claim 1 when said program is run on a computer.

34. [original] A computer program product comprising program code means stored on a computer readable medium for performing the method of Claim 1 when said program product is run on a computer.

35. [amended herein] A network element ~~(952)~~ for handling data packets, said data packets belonging to a set of data packets, said network element comprising

- means ~~(954)~~ for capturing data packets,

~~**characterized** in that it further comprises:~~

- means ~~(958)~~ for accepting a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet.

36. [amended herein] A network element ~~(952)~~ according to claim 35, ~~**characterized** in that it further comprises~~ further comprising:

- means ~~(956)~~ for processing a captured data packet, and
- means ~~(960)~~ for determining a modification command affecting at least one captured data packet, as a response to processing said at least one captured data packet.

37. [amended herein] A network element cluster ~~(950)~~ for handling data packets, said data packets belonging to a set of data packets, at least one node ~~(952)~~ of said network element cluster comprising

- means ~~(954)~~ for capturing data packets,

~~**characterized** in that it further comprises:~~

- means ~~(958)~~ for accepting a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet.

38. [amended herein] A network element cluster ~~(950)~~ according to claim 37, ~~**characterized** in that~~ wherein said at least one node ~~(952)~~ further comprises

- means ~~(962)~~ for maintaining a list of modification commands, said list enabling modification of captured data packets, and

- means ~~(964)~~ for transmitting said list of modification commands from said node to another node of said cluster of network elements.

39. [new] A storage medium carrying a computer-executable software entity for handling data packets, said data packets belonging to a set of data packets, said software entity comprising

- program code configured to capture data packets,
- program code configured to accepting a captured data packet for processing based on said captured data packet and data packets captured prior to said captured data packet,
- program code configured to, when processing at least one captured data packet accepted for processing, determine a modification command affecting at least said at least one captured data packet,
- program code configured to maintain a list of modification commands, said list enabling modification of captured data packets,
- program code configured to decline a captured data packet from processing, if said captured data packet is already processed and modification commands induced by said captured data packet are already determined and maintained in said list of modification commands,
- program code configured to modify said captured data packet based on said list of modification commands, and
- program code configured to release the modified captured data packet.

40. [new] A storage medium carrying a computer-executable software entity for

handling data packets, said data packets belonging to a set of data packets having at least partly hierarchical structure, said software entity comprising:

- program code configured to capture data packets,
- program code configured to accept a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet such that data packets are accepted for processing in the order specified by said at least partly hierarchical structure.

41. [new] A storage medium carrying a computer-executable software entity for handling data packets, said data packets belonging to a set of data packets and forming a plurality of groups of data packets, said software entity comprising:

- program code configured to capture data packets,
- program code configured to decline a captured data packet belonging to a first group of data packets from processing, if all other data packets belonging to said first group of data packets are not yet captured, and delaying said captured data packet, and
- program code configured to process a captured data packet belonging to a first group of data packets together with delayed data packets belonging to said first group of data packets, if said captured data packet belonging to said first group and said delayed data packets belonging to said first group form a full first group of data packets.

42. [new] A storage medium carrying a computer-executable software entity for handling data packets, said data packets belonging to a set of data packets and forming a plurality of groups of data packets, said software entity comprising:

- program code configured to capture data packets, said data packets containing information fragments belonging to a sequence of information fragments,

- program code configured to decline a captured data packet from processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is not yet captured, and
- program code configured to accept a captured data packet for processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is already processed, whereby each information fragment of said sequence is processed only once.

43. [new] A network element for handling data packets, comprising

- program code configuring the network element to capture data packets, said data packets belonging to a set of data packets,
- program code configuring the network element to accept a captured data packet for processing based on said captured data packet and data packets captured prior to said captured data packet,
- program code configuring the network element to, when processing at least one captured data packet accepted for processing, determine a modification command affecting at least said at least one captured data packet,
- program code configuring the network element to maintain a list of modification commands, said list enabling modification of captured data packets,
- program code configuring the network element to decline a captured data packet from processing, if said captured data packet is already processed and modification commands induced by said captured data packet are already determined and maintained in said list of modification commands,

- program code configuring the network element to modify said captured data packet based on said list of modification commands, and

- program code configuring the network element to release the modified captured data packet.

44. [new] A network element for handling data packets, comprising

- program code configuring the network element to capture data packets, said data packets belonging to a set of data packets having at least partly hierarchical structure,

- program code configuring the network element to accept a captured data packet for processing or declining a captured data packet from processing based on said captured data packet and data packets captured prior to said captured data packet such that data packets are accepted for processing in the order specified by said at least partly hierarchical structure.

45. [new] A network element for handling data packets, comprising

- program code configuring the network element to capture data packets, said data packets belonging to a set of data packets and forming a plurality of groups of data packets,

- program code configuring the network element to decline a captured data packet belonging to a first group of data packets from processing, if all other data packets belonging to said first group of data packets are not yet captured, and delaying said captured data packet, and

- program code configuring the network element to process a captured data packet belonging to a first group of data packets together with delayed data packets belonging to said first group of data packets, if said captured data packet belonging to said first group and said delayed data packets belonging to said first group form a full first group of data packets.

46. [new] A network element for handling data packets, comprising

- program code configuring the network element to capture data packets, said data packets belonging to a set of data packets and forming a plurality of groups of data packets and containing information fragments belonging to a sequence of information fragments,

- program code configuring the network element to decline a captured data packet from processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is not yet captured, and

- program code configuring the network element to accept a captured data packet for processing, if a data packet containing the information fragment immediately preceding the information fragments of said captured data packet in said sequence is already processed, whereby each information fragment of said sequence is processed only once.